

ЦИФРОВАЯ ПЛАТФОРМА ДЛЯ СУПЕРКОМПЬЮТЕРНОГО МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ ПРОЦЕССОВ НАПЫЛЕНИЯ

Н. И. Тарасов¹ [0000-0002-2342-1292], В. О. Подрыга² [0000-0001-7874-6978],

С. В. Поляков³ [0000-0003-1859-9034], А. В. Тимаков⁴ [0000-0001-6937-8018]

¹⁻⁴ *Институт прикладной математики им. М.В. Келдыша РАН, Миусская пл., 4, г. Москва, 125047*

¹nikita_tarasov@imamod.ru, ²pvictoria@list.ru, ³polyakov@imamod.ru,

⁴aletimakov@yandex.ru

Аннотация

Представлена цифровая платформа суперкомпьютерного моделирования в задачах напыления частиц на подложки. Целью работы является обсуждение общей архитектуры, технологического стека и особенностей реализации пользовательского интерфейса платформы. В ее основу положены веб-технологии доступа и управления расчетами, позволяющие реализовать пользовательскую систему для проведения полного цикла вычислительного эксперимента, включающего конфигурацию прикладных приложений, их запуск на удаленных вычислительных ресурсах, мониторинг прохождения задач, анализ и интерактивную визуализацию результатов. Взаимодействие пользователя с вычислительными ресурсами реализовано посредством графического интерфейса, не требующего от компьютера пользователя наличия дополнительного программного обеспечения, кроме веб-браузера. Важным преимуществом платформы является возможность проведения широкомасштабных компьютерных исследований в многопользовательском режиме, вытекающая из естественных принципов построения клиент-серверных приложений. Представляемая цифровая веб-платформа была успешно апробирована на вычислительных кластерах ИПМ им. М.В. Келдыша РАН при решении ряда актуальных математических задач нанотехнологии. Также с ее помощью последние 3 года проводится групповое обучение студентов МФТИ современным информационным технологиям.

Ключевые слова: суперкомпьютерное моделирование, цифровая платформа, веб-интерфейс, газодинамическое напыление частиц

ВВЕДЕНИЕ

Проблема напыления частиц на подложки входит в круг фундаментальных и прикладных задач нанотехнологии. В ее рамках актуально исследование математическими методами всех стадий процесса напыления, что позволит эффективно и экономно изготавливать наноматериалы и нанопокртия с заданными свойствами, используемые во многих отраслях промышленности. Вычислительное ядро, связанное с реализацией численных методов решения возникающих здесь математических задач, ориентировано изначально на применение мощных вычислительных кластеров и суперкомпьютеров. Оно предполагает работу исследователя с большими распределенными наборами данных, обработка которых также требует применения параллельных вычислений. В связи с этим целью настоящей работы является разработка специализированной веб-платформы для суперкомпьютерного моделирования процессов напыления в рамках различных физико-технических подходов.

Применение суперкомпьютерного моделирования в подобных исследованиях стало уже повсеместной практикой. Однако для достижения максимального эффекта от использования суперЭВМ необходима разработка многозадачных многопользовательских сред и сервисов. Последние в настоящее время разрабатываются в рамках облачных и веб-технологий. Практическая реализация таких сред предполагает создание удобного сетевого интерфейса пользователей и программных ядер, отвечающих за решение отдельных вычислительно емких задач прикладной направленности. Особое внимание при этом уделяется совместной работе пользователей над одним или несколькими проектами. В последнее время подобные программные среды получили название цифровых платформ, что связано с внедрением в них технологий искусственного интеллекта. В классическом CFD-моделировании эти технологии используются на этапе анализа полученных численных результатов большого объема, то есть больших данных.

АРХИТЕКТУРА ВЕБ-ПЛАТФОРМЫ

Выбирая концепцию создаваемой платформы, мы остановились на веб-решениях, которые в отличие от облачных технологий (имеется ввиду их вариант в виде инфраструктуры как услуги – IaaS) менее чувствительны к техническим параметрам конкретных компьютерных и суперкомпьютерных систем, менее затратные по срокам реализации и стоимости поддержки, но при этом поддерживают многоплатформенный и многопользовательский режимы работы. К тому же веб-технологии не исключают возможности использования систем облачных вычислений при наличии стандартного или специализированного интерфейса доступа и становятся некоторой единой точкой взаимодействия для конечного пользователя. При этом допускается использование кластеров и суперкомпьютеров, не интегрированных в облачные системы (таких, например, как гибридные суперкомпьютеры ИПМ им. М.В. Келдыша РАН К-60 или К-100 [1]).

Выполнение конкретных расчетных задач традиционно было предложено реализовать с помощью запуска приложений в пакетном режиме. Такой подход позволяет гибко настраивать работу конкретных расчетных программ посредством формирования необходимых конфигурационных файлов и скриптов, которые можно запускать в локальном или удаленном режимах с помощью ssh-консоли. Подобная технология, реализуемая непосредственно пользователем, требует от него определенных знаний и навыков работы с терминалом, а также самостоятельного учета, мониторинга и анализа всех запущенных или завершившихся расчетов. Последнее фактически делает работу пользователя неэффективной, а иногда и просто невозможной. Для решения этой технической проблемы и создается интерактивная и интеллектуальная в определенном смысле веб-среда (веб-платформа), которая принимает на себя максимальное количество административных, технических и прочих трудоемких функций, а также предоставляет пользователю дружественное визуальное общение с большим количеством программ, данных и коллег по проектам.

Предлагаемая нами цифровая веб-платформа, схема которой изображена на рис. 1, представляет собой универсальный инструмент взаимодействия разработчиков и пользователей прикладных программных кодов с доступными им аппаратными средствами (от персональных компьютеров до рабочих станций и

мощных вычислительных кластеров) посредством графического веб-интерфейса, поддерживающего следующие возможности:

- авторизация пользователей;
- проведение полного цикла компьютерного моделирования процесса напыления частиц, в том числе многопользовательского;
- возможность запуска заданий на удаленных суперкомпьютерных ресурсах;
- формирование базы данных расчетов и обеспечение доступа к их результатам;
- обеспечение возможностей анализа результатов.

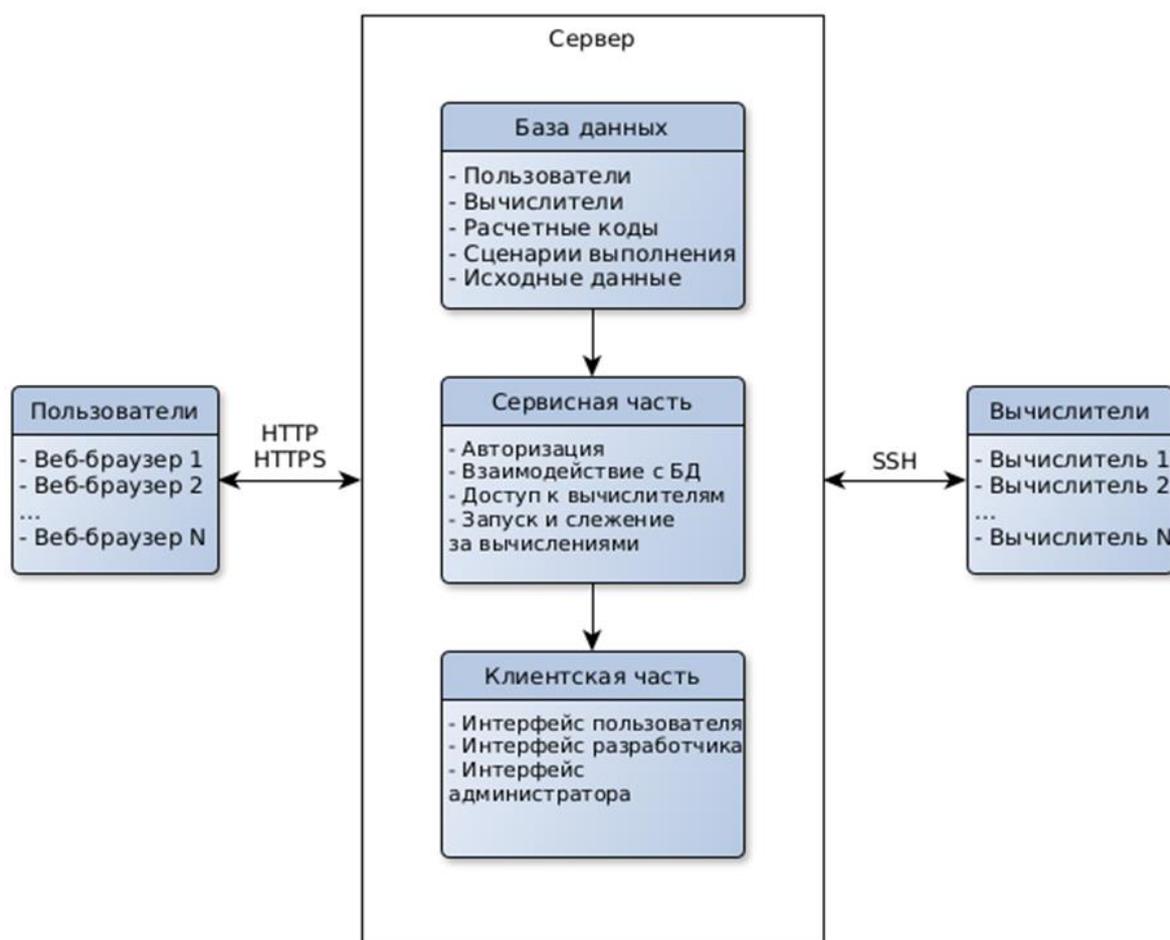


Рис. 1. Схема цифровой веб-платформы.

Взаимодействие пользователей между собой заключается в следующем. Во-первых, это может быть совместная разработка кодов приложений, для чего используются система контроля версий и общие папки на сервере проекта. Во-

вторых, это совместное проведение расчетов и обработка их результатов в рамках общего проекта. Такая возможность регулируется с помощью гибкой системы прав доступа к проекту и защиты от одновременного запуска расчетных программ с одинаковыми начальными данными из разных логинов. Отдельно отметим особую актуальность данного функционала, вытекающую из практики проведения комплексных вычислительных экспериментов в выбранной предметной области. Применяемый здесь многомасштабный подход при моделировании сложных физических процессов в широком диапазоне пространственных размеров (от метров до нанометров) опирается на синтез классических моделей механики сплошных сред, методов частиц и/или Монте-Карло и численный анализ свойств веществ и материалов на их основе. Очевидно, проведение вычислительного эксперимента подобного объема и качества силами одного специалиста представляет собой трудоемкую задачу. Однако при наличии возможности участия нескольких экспертов из различных прикладных областей, а также при использовании накопленных баз данных по свойствам веществ и материалов и решениям отдельных актуальных подзадач проведение комплексных исследований в рамках выполнения совместных расчетных проектов становится достижимым и весьма эффективным решением.

НАБОР ТЕХНОЛОГИЙ

Первая разработка предлагаемой цифровой платформы [2] основывалась на Django [3] – библиотеке для языка программирования Python, содержащей все необходимое для создания клиент-серверного решения, в том числе механизм объектно-реляционного отображения (ORM) для взаимодействия с базами данных, генератор веб-страниц из html-шаблонов, системы маршрутизации и авторизации. В процессе разработки выяснилось, что использования подхода шаблонных страниц недостаточно для создания интерактивного пользовательского интерфейса, обеспечивающего возможности, необходимые для проведения вычислительного эксперимента. Для поддержки требуемого функционала приходилось использовать JavaScript код, исполняемый непосредственно в веб-браузере. В итоге от использования шаблонов было решено отказаться в пользу разработки

одностраничного приложения (SPA) в реактивном стиле, для чего была применена библиотека Vue.js [4]. В данной разработке было решено исключить также и Django, поскольку большая часть его функционала, заключающаяся в шаблонах веб-страниц, перестала использоваться.

Программной основой для серверной части цифровой платформы стал Node.js [5], являющийся расширением JavaScript. Обычно язык программирования JavaScript используется на стороне веб-клиента и поддерживается во всех современных браузерах. Однако он имеет существенные ограничения по функциональности, в том числе невозможность взаимодействия с устройствами ввода-вывода и системными потоками напрямую. Для расширения JavaScript до языка общего назначения в целях создания серверного решения и используется Node.js.

Применение Node.js позволило, с одной стороны, добиться однородности кодовой базы между клиентской и серверной частями, а с другой – использовать высокую степень асинхронности JavaScript на сервере. Это особенно полезно для целей слежения за прохождением задач на удаленных вычислителях, выполнения длительных заданий и масштабирования системы.

Для обеспечения маршрутизации, а также использования плагинов безопасности был использован фреймворк Express.js [6], авторизация реализована посредством библиотеки passport.js [7]. Взаимодействие с базой данных осуществлялось с помощью библиотеки ORM Sequelize [8]. В качестве базы данных на стадии разработки использовалась SQLite [9].

Поскольку создаваемая цифровая платформа является закрытой для общего доступа, было решено организовать клиент-серверное взаимодействие на основе веб-сокетов [10]. Веб-сокеты представляют собой надстройку над протоколом HTTP. В отличие от исходного протокола они поддерживают соединение после начального клиентского запроса о подключении. При успешном установлении такового клиент и сервер могут произвольно обмениваться сообщениями. Сообщения, отправляемые посредством веб-сокета, включают в себя исключительно тело запроса, а HTTP-заголовок повторно не отправляется, что позволяет существенно оптимизировать клиент-серверное общение. Для организации веб-сокетов на сервере применялась библиотека ws [11]. Обеспечение HTTPS-протокола достигалось посредством обратного прокси сервера nginx[12].

Для использования конкретных расчетных приложений и их объединения в функциональные группы (сценарии) в качестве формата метаописания используются файлы текстового формата YAML [13]. Выбранный формат является, с одной стороны, удобным для пользовательского ручного формирования, с другой – поддерживается в стандартной библиотеке JavaScript для сериализации (преобразование разнородных данных в битовую последовательность) и десериализации (обратное преобразование).

Клиентская часть цифровой платформы была реализована с применением реактивной парадигмы, организованной с помощью использования фреймворка Vue.js и библиотеки компонентов Quasar [14]. Для SPA-маршрутизации применялось естественное в данном случае решение Vue-router [15], для организации централизованного хранилища данных использовалась Vuex [16] (хранение информации о пользователе, открытого веб-сокета, уведомлений). В отношении компонента анализа результатов расчетов использован ParaViewWeb [17, 18] с встраиванием клиентской части в реализации Visualizer.js [19]. Данное решение обеспечивает поддержку форматов VTK, возможность интерактивной серверной или клиентской отрисовки, в зависимости от аппаратного обеспечения на компьютере пользователя. ParaViewWeb поддерживает большое количество фильтров данных, сопоставимое со стандартным «толстым» приложением ParaView. Отдельным преимуществом ParaViewWeb является возможность реализации собственного клиента на стандартном стеке цифровой платформы, что может в будущем привести к отказу от встраивания клиентской части платформы в Visualizer.js.

СЕРВЕР ЦИФРОВОЙ ПЛАТФОРМЫ

Взаимодействие с сервером цифровой платформы основано на использовании графических и иных элементов пользовательского интерфейса, в результате взаимодействия с которыми веб-браузер отправляет на сервер соответствующий запрос. Маршрутизация запросов осуществляется посредством унифицированных указателей ресурса (URL) и имеет следующий вид:

- GET /auth/ – проверка актуальности сессии текущего пользователя;
- POST /auth/login/ – авторизация пользователя;
- POST /auth/logout/ – завершение сессии текущего пользователя;

- POST /auth/register/ – регистрация нового пользователя;
- GET /files/ – получение файла, расположенного в рабочей директории веб-сервера;
- GET /files/ssh/ – получение файла с удаленного вычислительного ресурса;
- POST /files/tmp/ – отправка пользовательского файла на сервер во временное хранилище;
- GET /api/ – при обращении открывает основное веб-сокет соединение, предназначенное для обеспечения взаимодействия между клиентом (веб-браузером) и сервером цифровой платформы;
- GET /terminal/ - при обращении открывает веб-сокет соединение с веб-браузером пользователя, а также устанавливает ssh-соединение с требуемым удаленным вычислителем. После успешной установки соединений сервер связывает веб-сокет с ssh-клиентом.

Как указано выше, основным каналом взаимодействия между веб-браузером компьютера пользователя и сервером цифровой платформы является веб-сокет, открываемый при обращении по URL-адресу /api/. Сообщения, отправляемые клиентом, должны содержать обязательное поле «url» для вызова корректного обработчика запроса и опциональное поле «data», содержащее добавочную информацию. Поле «url» должно представлять собой строку, во многом соответствующую стандартному URL-адресу. Ответ сервера отправляется в сообщении, содержащем поля «url», «body» и числовой код в поле «status».

Необходимость наличия в ответе поля «url» объясняется особенностью работы веб-сокета, заключающейся в отсутствии классической схемы: запрос клиента и последующий ответ сервера. При использовании соединения данного типа клиент может только лишь подписаться на сообщения, отправляемые сервером (установка обработчика onmessage). В рамках разработанного подхода единый обработчик входящих сообщений вызывает конкретные клиентские функции, установленные при отправке запроса на сервер, на основании поля «url» ответа. Существуют два типа устанавливаемых обработчиков – постоянные (используемые при реагировании на сообщения о новом уведомлении или сообщении, подписке на изменение статуса расчетного задания) и единоразовые (используемые

при запросе информации из базы данных, списка файлов на удаленном вычислителе).

Отметим, что масштабирование веб-сервера производится посредством создания подпроцессов средствами Node.js. При этом главный процесс обеспечивает задачи слежения за расчетными заданиями и состоянием вычислительных ресурсов. Процессы-потомки используются для обработки пользовательских запросов и обслуживания установленных веб-сокет соединений.

Для хранения и обеспечения пользовательской информации используется база данных, имеющая структуру, изображенную на рис. 2.

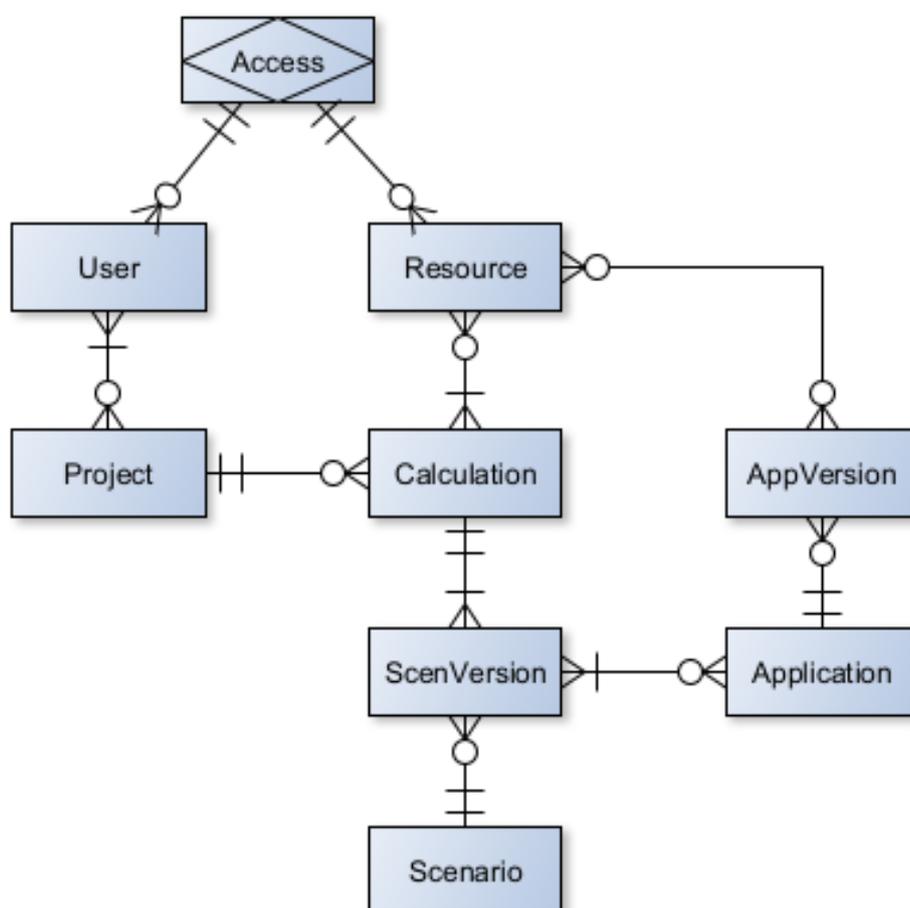


Рис. 2. Схема связей между компонентами базы данных.

Для поддержки возможности динамической регистрации удаленного вычислительного ресурса необходима информация об адресе вычислительного ресурса, порте ssh-сервера, системе прохождения задач, возможностях задания переменных окружения. Эта информация размещается в базе данных цифровой

платформы в таблице «Resource». Для осуществления ssh-доступа к удаленному вычислителю и разделения на нем пользователей используется таблица «Access», содержащая ссылки на записи пользователя системы и вычислителя, а также дополнительные переменные окружения и указание на используемые пользователем версии приложений. Подключение к удаленному вычислителю производится с использованием RSA-ключа веб-системы.

Описание расчетных приложений, помимо информационного, включает в себя также задание шаблонного метафайла, содержащего инструкции сборки, запуска, описание входных и выходных параметров, хранение которого производится в таблице «Application». Поддержка многообразия версий приложения достигается с помощью таблицы «AppVersion», содержащей дополнительно информацию о пути к исходному коду конкретной версии и итоговом метафайле, используемом для генерации графического пользовательского интерфейса.

Сценарии (таблица «Scenario») используются для объединения нескольких приложений в единый расчетный цикл. Они обеспечивают возможность запуска приложений и связывания их входных и выходных данных (с помощью файлового интерфейса и/или командной строки). Для описания сценариев используется файл определенной структуры. Поддержка многообразия версий сценариев производится при помощи таблицы «ScenVersion». Итоговое метаописание сценария используется при генерации графического пользовательского интерфейса для настройки прикладного ПО.

Суть каждого проекта платформы носит информационный характер, располагается в таблице «Project» и используется для объединения в единую смысловую конструкцию нескольких расчетов. Расчеты представляют собой циклы и результаты вычислений. Информация о них хранится в таблице «Calculation». Расчеты с точки зрения системы управления платформы являются основным способом взаимодействия с конкретным удаленным вычислителем. При этом под вычислителем понимается не только удаленный компьютер или кластер, но и конфигурация его использования (например, количество удаленных процессов, участвующих в расчете). При настройке конкретного расчета платформа генери-

рует графический интерфейс пользователя для ввода исходных данных, управления расчетным заданием (запуск, останов, продолжение, реконфигурация) и доступа к выходным данным на удаленном вычислителе.

ПОЛЬЗОВАТЕЛЬСКИЙ ВЕБ-ИНТЕРФЕЙС

Иллюстрацией реализации совместных расчетных проектов могут служить исследования ИПМ им. М.В. Келдыша РАН, ОИЯИ и ИПЛИТ РАН в рамках разработки программного комплекса GIMM_NANO [20]. Однако в этом решении еще не были задействованы в современном варианте ни облачные, ни веб-технологии. Тем не менее, объединение научных разработок вышеуказанных партнеров привело к существенному повышению качества научных исследований в области перспективных нанотехнологий.

Перейдем теперь к описанию работы пользователей в рамках предлагаемой цифровой платформы. После успешной авторизации пользователь, в соответствии с его ролью (администратор, разработчик или обычный пользователь), получает доступ к разделам системы, которые можно разбить на следующие функциональные группы:

- вычислители, доступные администраторам (см. рис. 3);
- приложения и сценарии, формируемые разработчиками (см. рис. 4);
- проекты и расчеты, используемые обычными пользователями (см. рис. 5).

Подобное распределение пользователей по ролям видится необходимым как для разграничения областей ответственности, так и для обеспечения максимально эффективной их работы в соответствии со специализацией. Так, основной деятельностью обычного пользователя являются проведение вычислений и анализ результатов компьютерного моделирования. При этом непосредственный доступ пользователя к кодам приложений, сценариям их работы и конкретным вычислителям ограничен возможностью получения справочной информации. Разработчики программного обеспечения заняты созданием новых и модификацией имеющихся приложений и сценариев, а также используют доступные вычислительные ресурсы с целью тестирования и адаптации кодов к тем или иным компьютерным архитектурам. Администраторы системы управляют аккаунтами поль-

зователей, регистрируют в системе новые вычислительные ресурсы, подготавливают их к проведению пользователями масштабных вычислительных экспериментов, обеспечивают целостность и безопасность платформы в целом. Если все три роли будут совмещаться «в одних руках», неизбежно начнутся проблемы деградации платформы ввиду нехватки нужных компетенций и времени на оперативное реагирование на возникающие нештатные ситуации.

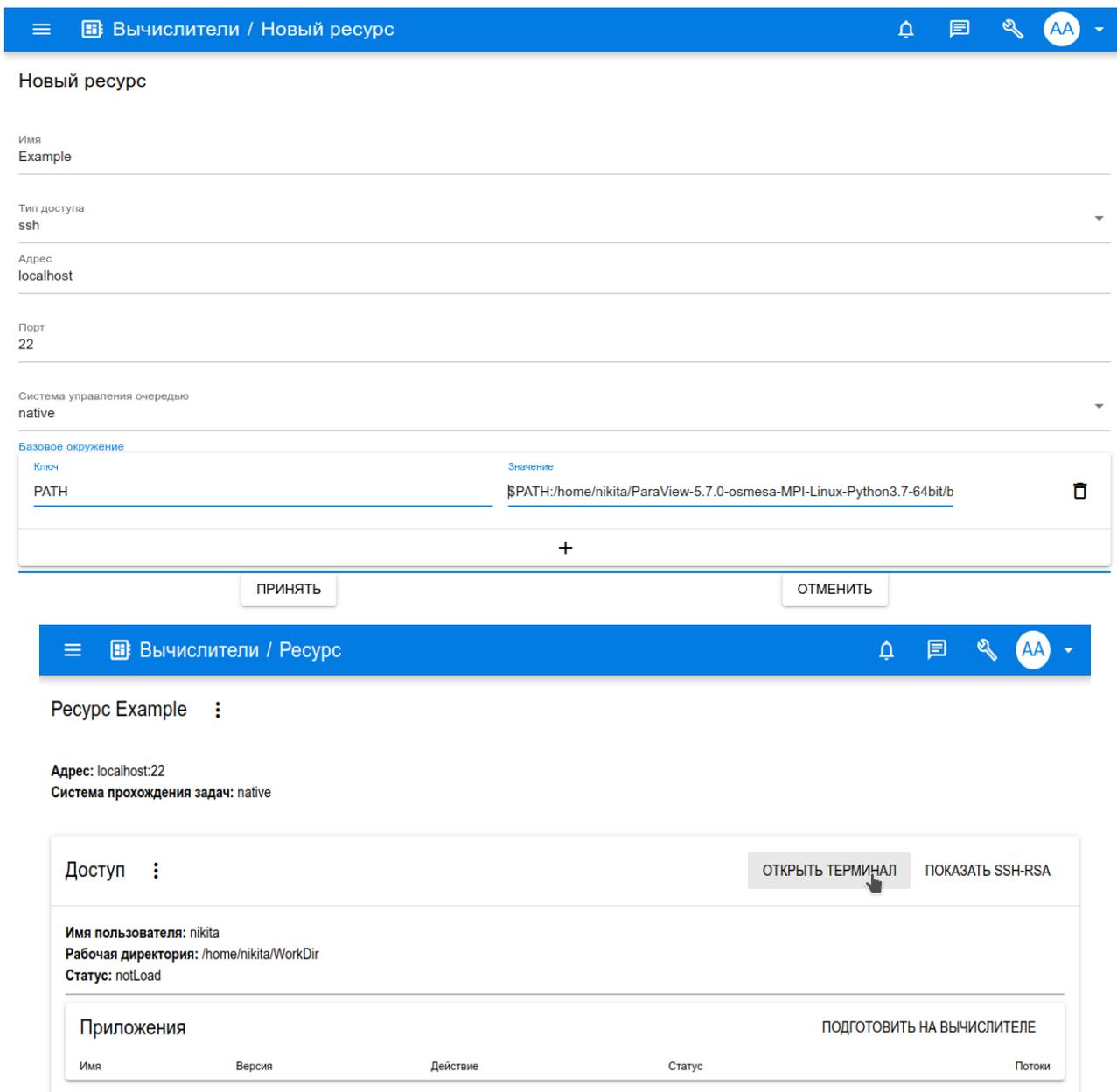


Рис. 3. Интерфейс администратора взаимодействия с вычислителями.

The screenshot displays the interface for developing and preparing applications. It is divided into two main sections.

Top Section: Passport Template Editor

The top part shows a code editor for a 'Паспорт' (Passport) template. The code is as follows:

```

1 build:
2   mkdir build; cd build; cmake ../; make -j4
3 exec:
4   comm: build/solver config.json
5 runBy:
6   type: exec
7 inputs:
8   - type: file
9     ident: mesh
10    name: Mesh file
11    descr: Mesh file in inner format
12   options:
13     format: data
14     path: mesh.txt
15   - type: file
16     ident: jsoncfg
17     name: Configuration
18     descr: Configuration file in json format
19   options:
20     format: tabs
21     vertical: true
22     fType: json
23     path: config1.json
24   childs:
25   - type: category
26     ident: main
27     name: Main
28     descr: Main global parameters
29   options:
30     format: form
31   childs:

```

Below the code editor is a progress bar for the source code upload, showing 'Исходный код' (Source code) at 111.9KB / 100.00%. A file named 'src.tar.gz' is listed as uploaded.

Bottom Section: Application Management

The bottom part shows the 'Приложение Solver' (Application Solver) management interface. It includes the following information:

- Идентификатор: solver
- Автор: Developer Developer
- Публичный: Да

There are three tabs: ОПИСАНИЕ, ШАБЛОН ПАСПОРТА, and ПОДГОТОВКА. The 'ПОДГОТОВКА' (Preparation) tab is active, showing a 'Ресурсы' (Resources) table with a 'ПОДГОТОВИТЬ НА ВЫЧИСЛИТЕЛЯХ' (Prepare on calculators) button.

Имя	Версия	Действие	Статус	Потоки
Example	V1	build	success	STDOUT STDERR

On the right, the 'Версии' (Versions) table is shown with a search bar and a 'НОВАЯ ЗАПИСЬ' (New Record) button.

Имя	Создан	Изменен	Действия
V1	04.09.2022, 13:26:04	04.09.2022, 13:26:04	🔗 ✎ 🗑

At the bottom right, it indicates 'Строк на странице: 5' (Rows per page: 5) and '1-1 из 1' (1 of 1).

Рис. 4. Интерфейс разработчика регистрации и подготовки приложений.

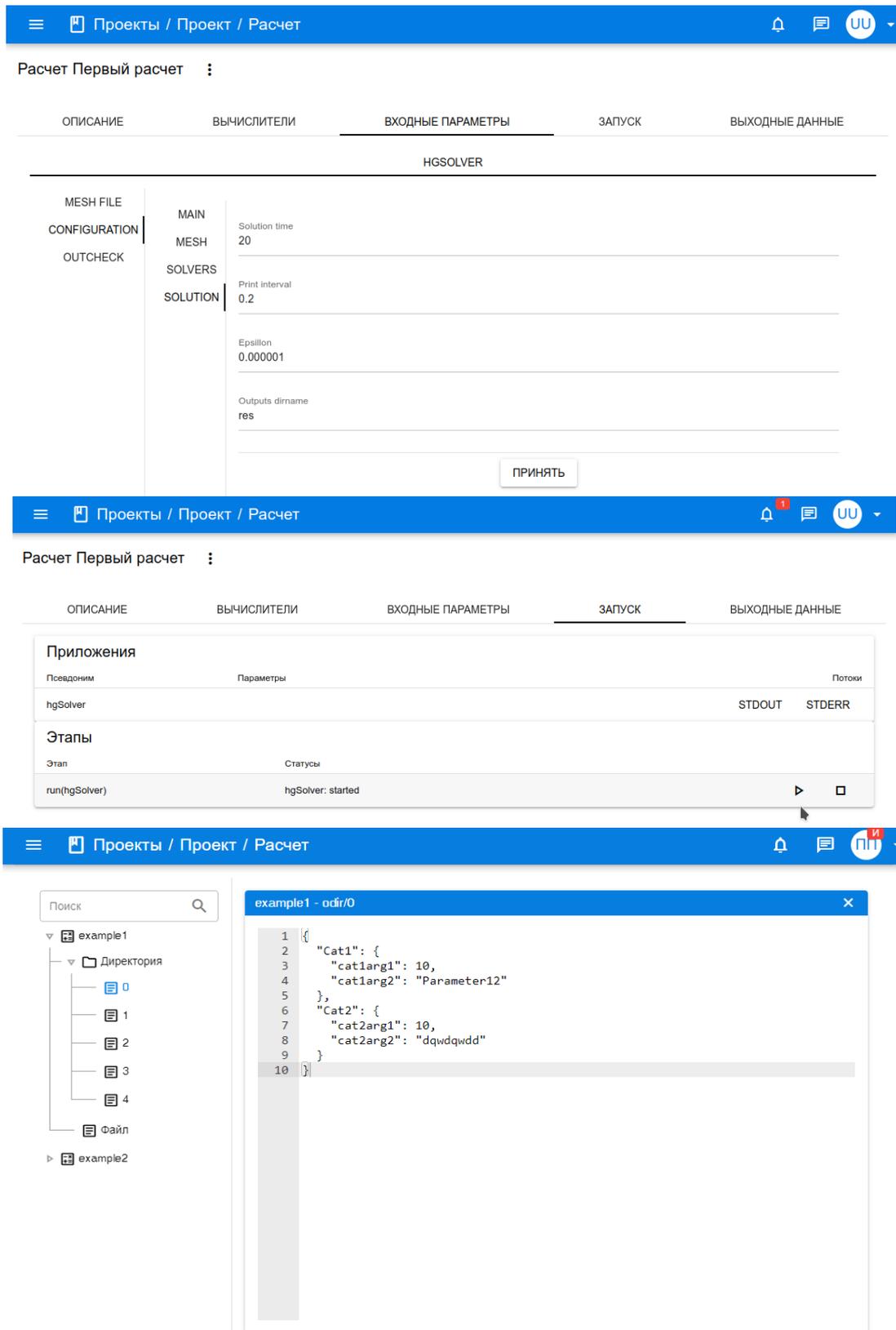


Рис. 5. Пользовательский интерфейс взаимодействия с расчетами.

Основой численного эксперимента в рамках платформы является расчет, при доступе к странице которого пользователю предоставляются графические формы выбора вычислительных ресурсов и задания исходных данных, компонента запуска и слежения, список результатов моделирования. Генерация интерфейса происходит в соответствии с выбранным сценарием на основе метаописания – текстового файла, содержащего список используемых приложений, а также связывание управляющих параметров, файлов данных, логику и порядок исполнения. Расчетные приложения платформы имеют метаописания, включающие в себя команды сборки и запуска, списки управляющих параметров, исходных данных и результатов исполнения.

Создание новых расчетов осуществляется на странице конкретного проекта, который позволяет каталогизировать и объединить некоторое их количество, задать общее описание.

В качестве примера работы первой версии веб-платформы приведем исследования в рамках проекта «Напыление». В его рамках возможно проведение расчетов нескольких прикладных задач. Одной из них является сверхзвуковое газодинамическое напыление отдельных металлических нанокластеров на подложку. В наших предыдущих работах по этой теме (см., например, [21, 22]) были представлены модель, численные алгоритмы, некоторые результаты конкретных параллельных вычислений и их анализа. Однако для выполнения всех вышеуказанных действий необходимо было программно реализовать сложную цепочку вычислительных экспериментов. Она начиналась с генерации исходных микро- и нанообъектов – например, столба азота микронных размеров и никелевых нанокластеров, находящихся в состоянии термодинамического равновесия. На основе этих данных генерировалось стартовое состояние системы газ–металл, в которой нанокластеры погружены в неподвижный азот. Затем производились разгон газовой среды и формирование ударной волны, которая заставляла кластеры двигаться в сторону подложки с нужной скоростью. На всех этапах такого моделирования использовались длительные параллельные вычисления. Переход от одного вычислительного эксперимента к другому производился пользователем «вручную». В частности, все расчеты сохранялись на дисках разных суперкомпьютеров, и их продолжение на очередной стадии требовало от пользователя переноса

больших данных с одной вычислительной системы на другую. Большой проблемой было отслеживание качества решения на всех стадиях вычислений ввиду отсутствия универсальных средств постобработки и визуализации распределенных данных большого объема. Частично эти и другие задачи были решены с помощью программных средств, разработанных нами ранее (см., [23, 24]). Однако полный цикл вычислений с минимальным участием пользователя стал доступным только в рамках разработки представляемой цифровой веб-платформы.

В частности, хотелось бы отметить в качестве нового программного решения визуализацию распределенных данных сверхбольшого объема. Оно стало возможным с появлением пакета ParaViewWeb, который пока не стал традиционным средством визуального анализа в отличие от его классической параллельной версии. В рамках реализации представляемой веб-платформы удалось интегрировать этот пакет в общую систему управления платформы и, тем самым, обеспечить максимальную полноту его функционала и управляемость со стороны пользователя.

В качестве примера на рис. 6 представлен этап визуализации рассчитанных на вычислительном кластере данных, представляющих собой распределения плотности газовой фазы в начальный и конечный моменты времени, описывающие процесс напыления на подложку.

Наиболее значимым в данном примере является то, что такая возможность визуализации позволяет анализировать результаты во время расчета на выбранном суперкомпьютере (например, можно контролировать состояние кластера и его маршрут, проверять целостность, изменение температуры, формы и т. д.). Для этого требуется только доступ системы управления платформой к дисковому пространству суперкомпьютера, на котором сохраняются результаты вычислений.

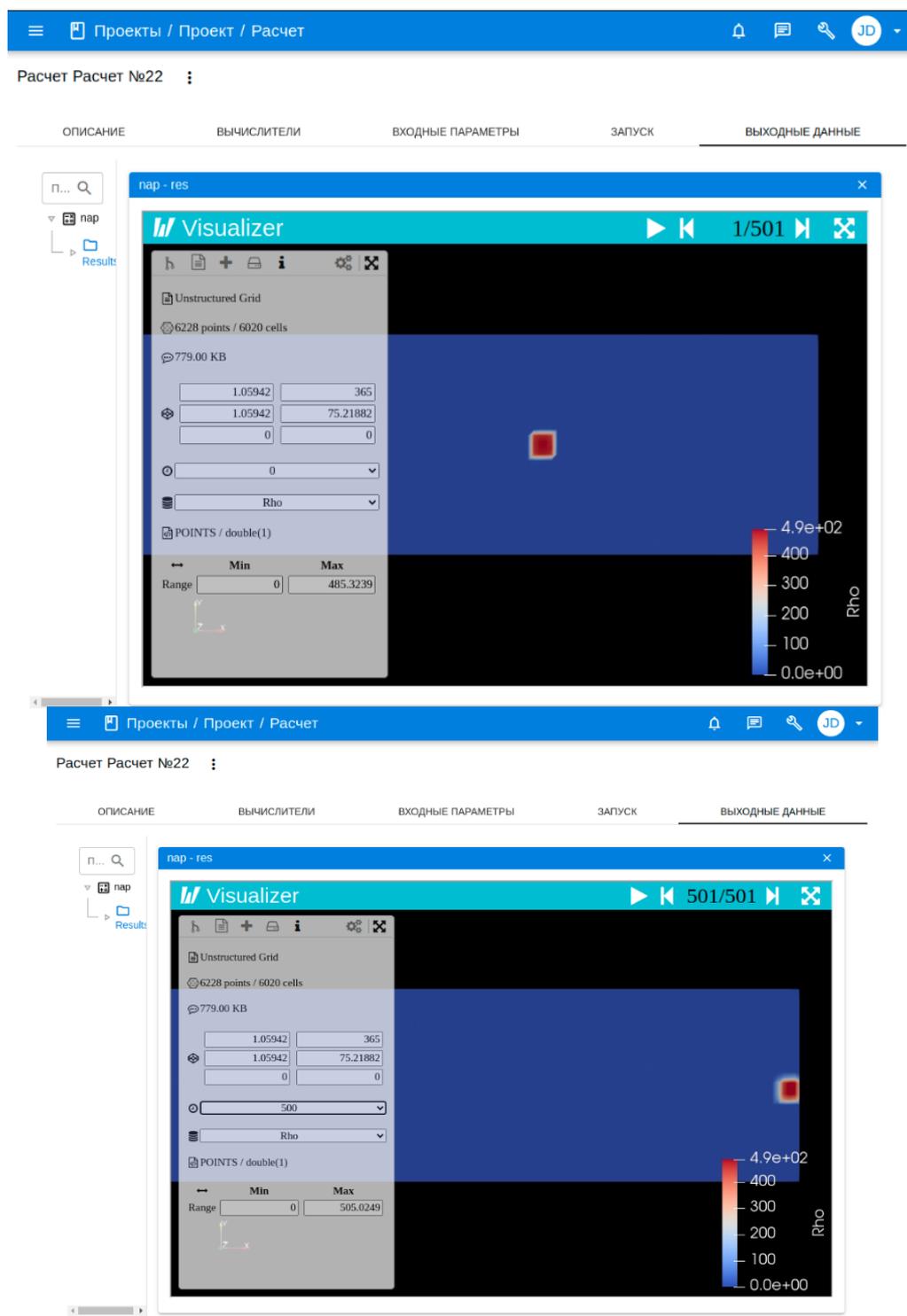


Рис. 6. Пример визуализации в рамках цифровой платформы.

Реализованное программное решение было апробировано на кластерах ИПМ им. М.В. Келдыша РАН при решении ряда задач вычислительной газо- и гидродинамики [25–28]. Эффективность его работы была подтверждена в период

пандемии COVID-19, когда практически все участники проекта (порядка 10 человек) работали удаленно. Также с помощью разработанной учебной версии веб-платформы уже третий год реализуется проект, связанный с групповым обучением студентов МФТИ современным информационным технологиям (количество студентов в группах 12–15 вместе с преподавателем). При этом ограничением производительности веб-платформы является конкретный ресурс задействованных серверов проекта. Например, обычный веб-сервер легко справляется с одновременной работой 100 и более пользователей, выполняя одновременно до 30000 запросов. В представляемом решении основная нагрузка приходится не столько на обработку веб-запросов, сколько на сценарии работы прикладных сервисов, связанных с пред- и постобработкой больших данных. Поскольку все эти задачи выполняются на удаленных вычислительных ресурсах, то большая часть нагрузки с клиентской части компьютера пользователя и веб-сервера управления платформой снимается. Исключением пока является просмотр множества видеопотоков, инициированных несколькими пользователями.

ЗАКЛЮЧЕНИЕ

Подводя итог, отметим, что на основе доработки ранее созданных пробных решений удалось реализовать новую версию веб-платформы, предназначенную для компьютерного и суперкомпьютерного моделирования задач напыления наночастиц на подложки и других связанных задач нанотехнологии. С ее помощью проведен ряд вычислительных экспериментов по напылению никелевых кластеров на подложки. В качестве суперкомпьютерных систем при этом были использованы гибридные суперкомпьютеры К60 и К100 и специализированные серверы ЦКП ИПМ им. М.В. Келдыша РАН. Количество пользователей цифровой платформы с учетом студенческой аудитории составило 25 чел. Отказов цифровой платформы при одновременной работе всех вышеуказанных пользователей не зафиксировано. В перспективе планируются расширение базы приложений, поддерживаемых платформой, и расширение функционала со стороны интерфейса пользователя. Планируется также поддержка альтернативных систем управления расчетными заданиями на удаленных вычислительных системах.

Благодарности

Работа выполнена при финансовой поддержке Российского научного фонда (проект № 21-71-20054).

СПИСОК ЛИТЕРАТУРЫ

1. Center of Collective Usage of KIAM RAS. URL: <https://ckp.kiam.ru/?home>
2. *Puzyrkov D.V., Podryga V.O., Polyakov S.V.* Cloud service for HPC management: ideas and appliance // *Lobachevskii Journal of Mathematics*. 2018. V. 39, No. 9. P. 1251–1261. <https://doi.org/10.1134/S1995080218090172>
3. The web framework for perfectionists with deadlines | Django. URL: <https://www.djangoproject.com/>
4. Vue.js–The Progressive JavaScript Framework | Vue.js. URL: <https://vuejs.org/>
5. About | Node.js. URL: <https://nodejs.org/en/about/>
6. Express – Node.js web application framework. URL: <https://expressjs.com/>
7. Passport.js. URL: <https://www.passportjs.org/>
8. Sequelize | Feature-rich ORM for modern TypeScript & JavaScript. URL: <https://sequelize.org/>
9. SQLite Home Page. URL: <https://sqlite.org/index.html>
10. RFC 6455 – The WebSocket Protocol. URL: <https://datatracker.ietf.org/doc/html/rfc6455>
11. ws: a Node.js WebSocket library. URL: <https://github.com/websockets/ws>
12. nginx. URL: <https://nginx.org/>
13. The Official YAML Web Site. URL: <https://yaml.org/>
14. Quasar Framework. URL: <https://quasar.dev/>
15. Home | Vue Router. URL: <https://router.vuejs.org/>
16. What is Vuex? | Vuex. URL: <https://vuex.vuejs.org/>
17. Web | ParaView. URL: <https://www.paraview.org/web/>
18. ParaViewWeb. URL: <http://kitware.github.io/paraviewweb/>
19. Visualizer. URL: <https://kitware.github.io/visualizer/>

20. Бондаренко А.А., Кононов Э.М., Косолапов О.А., Поляков С.В., Якобовский М.В. Программный комплекс GIMM_NANO // Научный сервис в сети Интернет: все грани параллелизма: Труды Международной суперкомпьютерной конференции (23–28 сентября 2013 г., г. Новороссийск). М.: Изд-во МГУ, 2013. С. 333–337. URL: <http://agora.guru.ru/abrau2013/pdf/333.pdf>

21. Поляков С.В., Подрыга В.О., Пузырьков Д.В., Кудряшова Т.А. Суперкомпьютерное молекулярное моделирование газодинамического напыления наночастиц на подложку // Суперкомпьютерные дни в России: Труды международной конференции (24–25 сентября 2018 г., г. Москва). М.: Изд-во МГУ, 2018. С. 782–792. URL: <http://russianscdays.org/files/pdf18/782.pdf>

22. Podryga V.O., Polyakov S.V. Multiscale mathematical modeling of the metal nanoparticles motion in a gas flow // In: Dimov I., Farago I., Vulkov L. (eds). Seventh International Conference on Finite Difference Methods. Theory and Applications. FDM 2018. Lecture Notes in Computer Science. Cham: Springer, 2019. V. 11386. P. 387–394. https://doi.org/10.1007/978-3-030-11539-5_44

23. Puzyrkov D., Podryga V., Polyakov S., Jakobovskii M. KIAM_JOB_CONTROL task management environment and its application to cloud and GRID computing) // Selected Papers of the 7th International Conference Distributed Computing and Grid-technologies in Science and Education (GRID 2016), ed. by V. Korenkov, T. Zaikina, F. Nechaevskiy, July 4–9, 2016, Dubna, Russia. CEUR Workshop Proceedings. 2017. V. 1787. P. 416–422. URL: <http://ceur-ws.org/Vol-1787/416-422-paper-72.pdf>

24. Kornilina M.A., Podryga V.O., Polyakov S.V., Puzyrkov D.V., Jakobovskiy M.V. Cloud service for solution of promising problems of nanotechnology // Supercomputing Frontiers and Innovations. 2017. V. 4, No. 4. P. 66–79. <https://doi.org/10.14529/jsfi170405>

25. Polyakov S.V., Podryga V.O., Kudryashova T.A. HPC simulation of non-linear processes in microsystems gas-metal // Lobachevskii Journal of Mathematics. 2020. V. 41, No. 8. P. 1554–1562. <https://doi.org/10.1134/S1995080220080168>

26. Podryga V.O., Polyakov S.V., Tarasov N.I. Developing of multiscale approach to HPC-simulation of multiphase fluid flows // Lobachevskii Journal of Mathematics. 2021. V. 42, No. 11. P. 2626–2636. <https://doi.org/10.1134/S1995080221110160>

27. Tarasov N., Karamzin Yu., Kudryashova T., Podryga V., Polyakov S. Computer modeling of air flows purification using sorption filters // Journal of Physics: Conf. Series. 2021. V. 2028. Paper 012025 (7 p.).

<https://doi.org/10.1088/1742-6596/2028/1/012025>

28. Kudryashova T.A., Polyakov S.V., Tarasov N.I. Mathematical modelling of electrophysical water treatment // Defect and Diffusion Forum. 2021. V. 412. P. 149–162.

<https://doi.org/10.4028/www.scientific.net/DDF.412.149>

DIGITAL PLATFORM FOR SUPERCOMPUTER MATHEMATICAL MODELING OF SPRAYING PROCESSES

N. I. Tarasov¹[0000-0002-2342-1292], V. O. Podryga²[0000-0001-7874-6978],

S. V. Polyakov³[0000-0003-1859-9034], A. V. Timakov⁴[0000-0001-6937-8018]

¹⁻⁴*Keldysh Institute of Applied Mathematics of RAS, Miusskaya sq., 4, Moscow, 125047*

¹nikita_tarasov@imamod.ru, ²pvictoria@list.ru, ³polyakov@imamod.ru,

⁴aletimakov@yandex.ru

Abstract

The work presents a digital platform for supercomputer modeling the problems of spraying the particles on substrates. The purpose of this work is to discuss the general architecture, technology stack and implementation features of the platform's user interface. The platform is based on web technologies for access and management of calculations, which allow implementing a user system for conducting a full cycle of a computational experiment, including the configuration of applied applications, their launch on remote computing resources, monitoring the completion of tasks, analysis and interactive visualization of results. User interaction with computing resources is implemented through the graphical interface that does not require the client computer to have any additional software, except actual version of a modern web browser. An important advantage of the platform is the ability to make large-scale computer research in a multi-user mode that is based on the natural principles of building client-server applications. The presented digital web platform was successfully tested on

computing clusters of the KIAM RAS in solving a number of the topical mathematical problems of nanotechnology. Also, with its help, for the last 3 years, group training of MIPT students in modern information technologies has been carried out.

Keywords: *supercomputer modeling, digital platform, web interface, gas-dynamic spraying of particles*

REFERENCES

1. Center of Collective Usage of KIAM RAS. URL: <https://ckp.kiam.ru/?home>
2. Puzyrkov D.V., Podryga V.O., Polyakov S.V. Cloud service for HPC management: ideas and appliance // Lobachevskii Journal of Mathematics. 2018. V. 39, No. 9. P. 1251–1261. <https://doi.org/10.1134/S1995080218090172>
3. The web framework for perfectionists with deadlines | Django.
URL: <https://www.djangoproject.com/>
4. Vue.js–The Progressive JavaScript Framework | Vue.js.
URL: <https://vuejs.org/>
5. About | Node.js. URL: <https://nodejs.org/en/about/>
6. Express – Node.js web application framework. URL: <https://expressjs.com/>
7. Passport.js. URL: <https://www.passportjs.org/>
8. Sequelize | Feature-rich ORM for modern TypeScript & JavaScript.
URL: <https://sequelize.org/>
9. SQLite Home Page. URL: <https://sqlite.org/index.html>
10. RFC 6455 – The WebSocket Protocol.
URL: <https://datatracker.ietf.org/doc/html/rfc6455>
11. ws: a Node.js WebSocket library. URL: <https://github.com/websockets/ws>
12. nginx. URL: <https://nginx.org/>
13. The Official YAML Web Site. URL: <https://yaml.org/>
14. Quasar Framework. URL: <https://quasar.dev/>
15. Home | Vue Router. URL: <https://router.vuejs.org/>
16. What is Vuex? | Vuex. URL: <https://vuex.vuejs.org/>
17. Web | ParaView. URL: <https://www.paraview.org/web/>
18. ParaViewWeb. URL: <http://kitware.github.io/paraviewweb/>
19. Visualizer. URL: <https://kitware.github.io/visualizer/>

20. Bondarenko A.A., Kononov E.M., Kosolapov O.A., Polyakov S.V., Yakobovskij M.V. Programmnyj kompleks GIMM_NANO // Nauchnyj servis v seti Internet: vse grani parallelizma: Trudy Mezhdunarodnoj superkomp'yuternoj konferencii (23–28 sentyabrya 2013 g., g. Novorossiysk). M.: Izd-vo MGU, 2013. S. 333–337. URL: <http://agora.guru.ru/abrau2013/pdf/333.pdf>

21. Polyakov S.V., Podryga V.O., Puzyr'kov D.V., Kudryashova T.A. Superkomp'yuternoe molekulyarnoe modelirovanie gazodinamicheskogo napyleniya nanostochastic na podlozhku // Superkomp'yuternye dni v Rossii: Trudy mezhdunarodnoj konferencii (24–25 sentyabrya 2018 g., g. Moskva). M.: Izd-vo MGU, 2018. S. 782–792. URL: <http://russianscdays.org/files/pdf18/782.pdf>

22. Podryga V.O., Polyakov S.V. Multiscale mathematical modeling of the metal nanoparticles motion in a gas flow // In: Dimov I., Farago I., Vulkov L. (eds). Seventh International Conference on Finite Difference Methods. Theory and Applications. FDM 2018. Lecture Notes in Computer Science. Cham: Springer, 2019. V. 11386. P. 387–394. https://doi.org/10.1007/978-3-030-11539-5_44

23. Puzyr'kov D., Podryga V., Polyakov S., Yakobovskii M. KIAM_JOB_CONTROL task management environment and its application to cloud and GRID computing) // Selected Papers of the 7th International Conference Distributed Computing and Grid-technologies in Science and Education (GRID 2016), ed. by V. Korenkov, T. Zaikina, F. Nechaevskiy, July 4–9, 2016, Dubna, Russia. CEUR Workshop Proceedings. 2017. V. 1787. P. 416–422. URL: <http://ceur-ws.org/Vol-1787/416-422-paper-72.pdf>

24. Kornilina M.A., Podryga V.O., Polyakov S.V., Puzyr'kov D.V., Yakobovskiy M.V. Cloud service for solution of promising problems of nanotechnology // Supercomputing Frontiers and Innovations. 2017. V. 4, No. 4. P. 66–79. <https://doi.org/10.14529/jsfi170405>

25. Polyakov S.V., Podryga V.O., Kudryashova T.A. HPC simulation of non-linear processes in microsystems gas-metal // Lobachevskii Journal of Mathematics. 2020. V. 41, No. 8. P. 1554–1562. <https://doi.org/10.1134/S1995080220080168>

26. Podryga V.O., Polyakov S.V., Tarasov N.I. Developing of multiscale approach to HPC-simulation of multiphase fluid flows // Lobachevskii Journal of Mathematics. 2021. V. 42, No. 11. P. 2626–2636. <https://doi.org/10.1134/S1995080221110160>

27. Tarasov N., Karamzin Yu., Kudryashova T., Podryga V., Polyakov S. Computer modeling of air flows purification using sorption filters // Journal of Physics: Conf. Series. 2021. V. 2028. Paper 012025 (7 p.).

<https://doi.org/10.1088/1742-6596/2028/1/012025>

28. Kudryashova T.A., Polyakov S.V., Tarasov N.I. Mathematical modelling of electrophysical water treatment // Defect and Diffusion Forum. 2021. V. 412. P. 149–162.

<https://doi.org/10.4028/www.scientific.net/DDF.412.149>

СВЕДЕНИЯ ОБ АВТОРАХ



ТАРАСОВ Никита Игоревич – исследователь, младший научный сотрудник ИПМ им. М.В. Келдыша РАН.

Nikita Igorevich TARASOV – researcher, junior researcher of KIAM RAS.

email: nikita_tarasov@imamod.ru

ORCID 0000-0002-2342-1292



ПОДРЫГА Виктория Олеговна – доктор физико-математических наук, доцент, ведущий научный сотрудник ИПМ им. М.В. Келдыша РАН.

Viktoriia Olegovna PODRYGA – Doctor of Physics and Mathematics, Associate Professor, Leading Researcher of KIAM RAS.

email: pvictoria@list.ru

ORCID 0000-0001-7874-6978

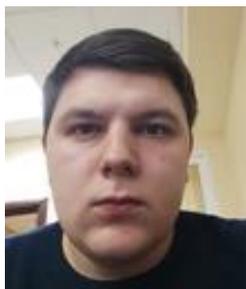


ПОЛЯКОВ Сергей Владимирович – доктор физико-математических наук, старший научный сотрудник, ведущий научный сотрудник ИПМ им. М.В. Келдыша РАН.

Sergey Vladimirovich Polyakov – Doctor of Physics and Mathematics, Associate Professor, Leading Researcher of KIAM RAS.

email: polyakov@imamod.ru

ORCID 0000-0003-1859-9034



ТИМАКОВ Алексей Валерьевич – аспирант МФТИ (ГУ), инженер ИПМ им. М.В. Келдыша РАН

Alexey Valerievich TIMAKOV – Post-graduate student of the Moscow Institute of Physics and Technology (GU), engineer of KIAM RAS

email: aletimakov@yandex.ru

ORCID 0000-0001-6937-8018

Материал поступил в редакцию 15 декабря 2022 года